

(19) 日本国特許庁(JP)

(12) **公開特許公報(A)**

(11) 特許出願公開番号

特開2005-242995

(P2005-242995A)

(43) 公開日 平成17年9月8日(2005.9.8)

(51) Int.Cl.⁷

G O 6 F 12/00

G O 6 F 17/30

F 1

GO 6 F 12/00

GO 6 F 17/30

5 1 3 A

240C

テーマコード (参考)

5 B 0 7 5

5 B 082

審査請求 未請求 請求項の数 34 O L 外国語出願 (全 38 頁)

(21) 出願番号 特願2005-7069 (P2005-7069)
(22) 出願日 平成17年1月14日 (2005. 1. 14)
(31) 優先権主張番号 04075167.9
(32) 優先日 平成16年1月19日 (2004. 1. 19)
(33) 優先権主張国 欧州特許庁 (EP)

(71) 出願人 504189748
 エスエイピー エイジー
 SAP AG
 ドイツ、69190 ワルドルフ、ニュー
 ロットストラッセ 16
 Neurottstrasse 16, 6
 9190 Walldorf, Germa
 ny

(74) 代理人 100092897
 弁理士 大西 正悟

(74) 代理人 100115200
 弁理士 山口 修之

(72) 発明者 ウェルナー ホースト
 ドイツ、レッティゲイム 69242、リ
 ンデンヴェグ 18

最終頁に続く

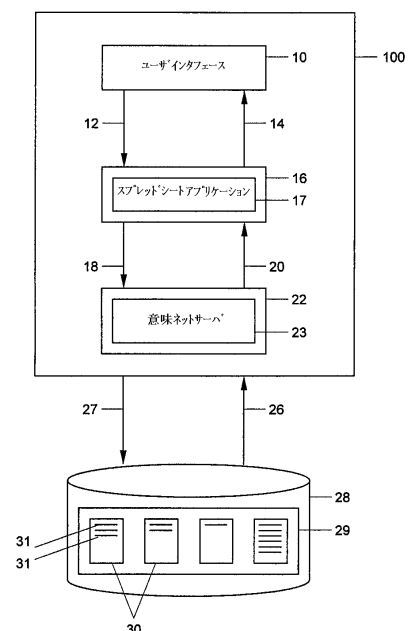
(54) 【発明の名称】 データベース管理システムおよびデータベース管理方法

(57) 【要約】 (修正有)

【課題】 データベース管理システムおよびデータベース管理方法を提供する。

【解決手段】データを記憶するためのデータベースを含むデータベースレイヤ28と、ビューに従ってデータベースレイヤからのデータをスプレッドシート内で表現するためのスプレッドシートレイヤとを備え、データベースレイヤとスプレッドシートレイヤとをインタフェースするように構成されたインタフェースレイヤ22を備え、データベースレイヤはそれぞれ複数のテーブル内の複数のエンティティタイプの複数のエンティティに対応するデータをデータベースに記憶し、複数のエンティティは所定の構造内の複数の関係によりリンクされており、インタフェースレイヤは、データベースレイヤ内のデータの構造を意味ネットに変換するための意味ネットサーバ23を備え、ビューは、データベースレイヤとスプレッドシートレイヤ間の相互運用性を容易にするために、意味ネットに関して定義される。

【選択図】 図1



【特許請求の範囲】

【請求項 1】

データを記憶するためのデータベースを含むデータベースレイヤ（28）と、ビューに従って前記データベースレイヤ（28）からのデータをスプレッドシート内で表現するためのスプレッドシートレイヤ（16）とを備えたデータベース管理システムであって、ここで、前記ビューは、前記データベースレイヤから前記スプレッドシートレイヤへのデータの配送、および前記スプレッドシートレイヤから前記データベースレイヤへのデータの保持のうちの、少なくとも1つを発生させ、そして、前記システムはさらに、前記データベースレイヤ（28）と前記スプレッドシートレイヤ（16）とをインタフェースするように構成されたインタフェースレイヤ（22）を備え、ここで、前記データベースレイヤ（28）は、それぞれ複数のテーブル内の複数のエンティティタイプの複数のエンティティに対応するデータを、前記データベースに記憶し、ここで、前記複数のエンティティは、所定の構造内の複数の関係によりリンクされており、ここで、前記インタフェースレイヤ（22）は、前記データベースレイヤ（28）内の前記データの前記構造を意味ネットに変換するための意味ネットサーバ（23）を備え、前記ビューは、前記データベースレイヤ（28）と前記スプレッドシートレイヤ（16）との間の相互運用性を容易にするために、前記意味ネットに関して定義される、データベース管理システム。

【請求項 2】

前記ビューが、前記スプレッドシート内の前記データを変更させた場合、前記データの任意の変更が前記データベースレイヤ（28）内に記憶される、請求項 1 に記載のデータベース管理システム。

【請求項 3】

前記意味ネットサーバ（23）は、前記データベースレイヤ（28）内の前記データ（31）の間の関係の意味ネットを記憶する、請求項 1 または 2 に記載のデータベース管理システム。

【請求項 4】

前記複数のテーブルの間の関係は、複数の外部キー参照として記憶される、請求項 1 ～ 3 のいずれか一項に記載のデータベース管理システム。

【請求項 5】

前記ビュー（12）は、前記インタフェースレイヤ（22）を介して、前記データベースレイヤ（28）内の前記データ（31）に対して実行され、それにより、データが前記スプレッドシートレイヤに配送され、所定の構造で、前記スプレッドシート（14）内で表現される、請求項 1 ～ 4 のいずれか一項に記載のデータベース管理システム。

【請求項 6】

前記意味ネットサーバ（23）は、前記データベース（29）の構造（30）から、意味ネット内の複数の項および関係タイプへの、マッピングを記憶する、請求項 1 ～ 5 のいずれか一項に記載のデータベース管理システム。

【請求項 7】

前記意味ネットサーバ（23）は、前記データベース（29）のコンテンツを表す前記データ（31）から、前記複数の項にそれぞれ対応する複数の副項への、マッピングを記憶する、請求項 6 に記載のデータベース管理システム。

【請求項 8】

前記ビューは、前記意味ネットの前記複数の項および関係タイプに関して定義される、請求項 6 または 7 に記載のデータベース管理システム。

【請求項 9】

前記意味ネットサーバ（23）は、前記意味ネット内で、前記データベースレイヤ（28）の前記データコンテンツを複製するように構成されている、請求項 1 ～ 8 のいずれか一項に記載のデータベース管理システム。

【請求項 10】

前記インタフェースレイヤ（22）は、前記ビューを解釈し、前記ビューに応答して、

前記意味ネットからの特定のデータを識別するように構成されている、請求項 1～9 のいずれか一項に記載のデータベース管理システム。

【請求項 1 1】

前記意味ネットサーバ (2 3) は、前記データベースレイヤ (2 8) の前記データコンテンツ (3 1) を間欠的に複製するように構成されている、請求項 9 または 1 0 に記載のデータベース管理システム。

【請求項 1 2】

前記インタフェースレイヤ (2 2) は、前記ビュー (1 8) を、データベースサーバ互換ビュー (2 7) に変換するように構成され、ここで、前記データベースサーバ互換ビューは、前記データベース (2 9) から、前記ビューに従ってデータ (3 1) を収集する、請求項 1～1 1 のいずれか一項に記載のデータベース管理システム。

10

【請求項 1 3】

前記意味ネットサーバ (2 3) は、前記収集されたデータを、前記意味ネットの前記項および関係にマッピングするように構成され、ここで、前記マッピングされたデータは前記スプレッドシート内に含まれる、請求項 1 2 に記載のデータベース管理システム。

【請求項 1 4】

前記スプレッドシート表現は、少なくとも 1 つのタイトル行を含み、ここで、前記ビューは、前記タイトル行にビューを入力することにより定義され、ここで、前記タイトル行は、それぞれ特定の関係に関連する複数のセルのうちの少なくとも 1 つを含み、ここで、前記ビューの前記関係は、1 つまたは複数の外部キー参照に対応し、前記ビューの意味項は、1 つまたは複数のデータベースエンティティタイプに対応している、請求項 1～1 3 のいずれか一項に記載のデータベース管理システム。

20

【請求項 1 5】

前記ビューに従った問合せが、データを収集するために前記データベース (2 9) に対して実行され、それらのデータは、所定の構造に従って、前記スプレッドシートレイヤにより生成されたスプレッドシート内にインポートされる、請求項 1～1 4 のいずれか一項に記載のデータベース管理システム。

【請求項 1 6】

前記スプレッドシートレイヤにより生成されたスプレッドシート内で表現された前記データは、前記インタフェースレイヤ (2 2) を介して、前記データベース (2 9) にインポートされる、請求項 1～1 5 のいずれか一項に記載のデータベース管理システム。

30

【請求項 1 7】

データベースを管理する方法であって、前記方法は、データベースレイヤ (2 8) 内のデータベース (2 9) 内にデータを記憶するステップと、ビューに従って前記データベースレイヤからのデータをスプレッドシート内で表現するステップとを含み、ここで、前記ビューは、前記データベースレイヤから前記スプレッドシートレイヤへのデータの配送、および前記スプレッドシートレイヤから前記データベースレイヤへのデータの保持のうちの、少なくとも 1 つを発生させ、前記方法はさらに、

40

前記データベースレイヤ (2 8) と前記スプレッドシートレイヤ (1 6) とをインタフェースするステップを含み、ここで、前記データベースレイヤ (2 8) は、それぞれ複数のテーブル内の複数のエンティティタイプの複数のエンティティに対応するデータを、前記データベースに記憶し、ここで、前記複数のエンティティは、所定の構造内の複数の関係によりリンクされており、ここで、前記インタフェースするステップは、前記データベースレイヤ (2 8) 内の前記データの構造を意味ネットに変換するステップを含み、前記表現するステップは、前記データベースレイヤ (2 8) と前記スプレッドシートレイヤ (1 6) との間の相互運用性を容易にするために、前記意味ネットに関して定義された前記ビューを入力するステップを含む、データベースを管理する方法。

【請求項 1 8】

50

入力により前記スプレッドシート内の前記データを変更させた場合は、前記スプレッドシート内の前記データの任意の変更が、前記インタフェースレイヤ（２２）を介して前記データベースレイヤ内に記憶される、請求項１７に記載の方法。

【請求項１９】

前記データベースレイヤ（２８）内の前記データ（３１）間の関係の意味ネットを記憶するように、前記意味ネットサーバ（２３）を構成するステップを含む、請求項１７または１８に記載の方法。

【請求項２０】

異なるそれぞれのテーブル内の異なるエンティティに属するデータを記憶し、エンティティ間の関係を外部キー参照によって記憶するように、前記データベースレイヤ（２８）を構成するステップを含む、請求項１７～１９のいずれか一項に記載の方法。 10

【請求項２１】

データが前記スプレッドシートレイヤに配送され、所定の構造で前記スプレッドシート内で表現されるように、前記インタフェースレイヤ（２２）を介して、前記データベースレイヤ（２８）内の前記データに対する前記ビューを実行するステップを含む、請求項１７～２０のいずれか一項に記載の方法。

【請求項２２】

前記データベースの構造から、意味ネット内の複数の項および関係タイプへのマッピングを記憶するように、前記意味ネットサーバ（２３）を構成する、さらなるステップを含む、請求項１７～２１のいずれか一項に記載の方法。 20

【請求項２３】

前記データベース（２９）のコンテンツを表すデータから、前記複数の項にそれぞれ対応する複数の副項へのマッピングを記憶するように、前記意味ネットサーバ（２３）を構成する、さらなるステップを含む、請求項２２に記載の方法。

【請求項２４】

前記意味ネットの前記複数の項および関係タイプに関して前記ビューを定義する、さらなるステップを含む、請求項２２または２３に記載の方法。

【請求項２５】

前記データベースレイヤ（２８）の前記データコンテンツを前記意味ネット内で複製するように、前記意味ネットサーバ（２３）を構成する、さらなるステップを含む、請求項１７～２４のいずれか一項に記載の方法。 30

【請求項２６】

前記ビューを解釈し、前記ビューに従って前記意味ネットからの特定のデータを識別するように、前記インタフェースレイヤ（２２）を構成するステップを含む、請求項１７～２５のいずれか一項に記載の方法。

【請求項２７】

前記データベース（２９）の前記コンテンツを間欠的に複製するように、前記意味ネットサーバ（２３）を構成する、さらなるステップを含む、請求項２５または２６に記載の方法。

【請求項２８】

前記ビューをデータベースサーバ互換ビュー（２７）に変換するように前記インタフェースレイヤ（２２）を構成する、さらなるステップを含み、ここで、前記データベースサーバ互換ビューは、前記データベース（２９）から前記データを収集するために使用される、請求項１７～２７のいずれか一項に記載の方法。 40

【請求項２９】

前記収集されたデータを前記意味ネットの前記項および関係にマッピングするように、前記意味ネットサーバ（２３）を構成する、さらなるステップを含み、ここで、前記マッピングされたデータは前記表現内に含まれる、請求項２８に記載の方法。

【請求項３０】

前記スプレッドシート表現は少なくともタイトル行を含み、そして、前記タイトル行内 50

に前記ビューを入力することにより前記ビューを定義するステップを含み、ここで、前記タイトル行は、それぞれ特定の関係に関連する複数のセルのうちの少なくとも1つを含み、ここで、前記ビューの前記関係は、1つまたは複数の外部キー参照に対応し、前記ビューの意味項は、1つまたは複数のデータベースエンティティタイプに対応している、請求項17～29のいずれか一項に記載の方法。

【請求項31】

データを収集するために、前記データベース（29）に対して、前記ビューに従った問合せを実行するさらなるステップを含み、ここで、前記収集されたデータは、所定の構造に従って、前記スプレッドシートレイヤ（16）により生成されたスプレッドシート表現内にインポートされる、請求項17～30のいずれか一項に記載の方法。

10

【請求項32】

スプレッドシート内で表現された前記データを、前記インタフェースレイヤ（22）を介して、前記データベースレイヤ（28）にインポートするステップを含む、請求項17～31のいずれか一項に記載の方法。

【請求項33】

請求項17～32のいずれか一項に記載の方法を実行するために動作可能な手段を含む、ユーザ端末。

【請求項34】

処理装置によって読み取り可能なプログラム記憶装置であって、前記プログラム記憶装置は、請求項17～32のいずれか一項に記載のステップを実行するために前記処理装置により実行可能な命令のプログラムを実施する、プログラム記憶装置。

20

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、データベース管理システムおよびデータベース管理方法に関する。

【背景技術】

【0002】

データベース、およびデータ用のその他のリポジトリは、データを記憶するためによく知られている。さらに、スプレッドシートまたは類似した表現の中でデータを表示するためのアプリケーションも公知である。そのようなアプリケーションは、以下、表現アプリケーションと記載する。

30

【発明の開示】

【発明が解決しようとする課題】

【0003】

従来の表現アプリケーションの1つの問題は、例えば構造化照会言語（SQL）などの、関連するデータベース照会言語と、データベースの内部データ構造との詳細な知識がなければ、ユーザが、データベースのコンテンツをスプレッドシートに転送すること、およびその逆は、不可能なことである。従来、データベースコンテンツについての、所望されるビューのための、構造化照会言語（SQL）問合せが定義されている。さらに、ビューコンテンツからスプレッドシートを作成する、ハードコード化プログラムも規定されている。そのようなビューを定義するには、SQLの知識とデータベースの内部データ構造の知識が必要とされるため、一般的なユーザがそのようなビューを定義することはできない。その上、さまざまなデータベースまたはその他のリポジトリからのデータを統合するビューの作成は、非常に複雑である。

40

【0004】

本発明の目的は、従来のデータベース、およびデータベースアプリケーションの部分である表現アプリケーションまたはスプレッドシートレイヤで発生するこれらの問題に対処することである。簡単にするために、他のアプリケーション内に統合された、独立したスプレッドシートアプリケーションおよびスプレッドシートレイヤの両方を、以下、スプレッドシートレイヤと記載する。

50

【0005】

特に、本発明の目的は、データベースコンテンツをスプレッドシート表現に転送すること、および、あるいは、スプレッドシート内で行われた変更を元のデータベースに組み込むことである。

【課題を解決するための手段】

【0006】

本発明の第1の態様によれば、データを記憶するためのデータベースを含むデータベースレイヤと、ビューに従ってデータベースレイヤからのデータをスプレッドシート内で表現するためのスプレッドシートレイヤとを備えたデータベース管理システムが提供され、ここで、ビューは、データベースレイヤからスプレッドシートレイヤへのデータの配送、およびスプレッドシートレイヤからデータベースレイヤへのデータの保持のうちの、少なくとも1つを発生させ、そして、前記システムはさらに、データベースレイヤとスプレッドシートレイヤとをインタフェースするように構成されたインタフェースレイヤを備え、ここで、データベースレイヤは、それぞれ複数のテーブル内の複数のエンティティタイプの複数のエンティティに対応するデータを、データベースに記憶し、ここで、複数のエンティティは、所定の構造内の複数の関係によりリンクされており、ここで、インタフェースレイヤは、データベースレイヤ内のデータの構造を意味ネットに変換するための意味ネットサーバを備え、ビューは、データベースレイヤとスプレッドシートレイヤとの間の相互運用性を容易にするために、意味ネットに関して定義される。

10

【0007】

データベースコンテンツの上に意味ネットインフラストラクチャを提供することにより、本発明は、ユーザがデータベース構造または問合せ言語の知識を有する必要なしに、データベースコンテンツが、例えばスプレッドシートなどの表現に転送されることを可能にする。本発明のさらなる利点は、ビューと結果との交換はクリップボードを介して完全に行われてもよい場合、スプレッドシートレイヤが変更される必要はないということである。

20

【0008】

本発明の第2の態様によれば、データベースを管理する方法が提供され、前記方法は、データベースレイヤ内のデータベース内にデータを記憶するステップと、ビューに従ってデータベースレイヤからのデータをスプレッドシート内で表現するステップとを含み、ここで、ビューは、データベースレイヤからスプレッドシートレイヤへのデータの配送、およびスプレッドシートレイヤからデータベースレイヤへのデータの保持のうちの、少なくとも1つを発生させ、前記方法はさらに、データベースレイヤとスプレッドシートレイヤとをインタフェースするステップを含み、ここで、データベースレイヤは、それぞれ複数のテーブル内の複数のエンティティタイプの複数のエンティティに対応するデータを、データベースに記憶し、ここで、複数のエンティティは、所定の構造内の複数の関係によりリンクされており、ここで、前記インタフェースするステップは、データベースレイヤ内のデータの構造を意味ネットに変換するステップを含み、前記表現するステップは、データベースレイヤとスプレッドシートレイヤとの間の相互運用性を容易にするために、意味ネットに関して定義されたビューを入力するステップを含む。

30

40

【0009】

本発明の第3の態様によれば、本発明の第1の態様の任意の方法を実行するために動作可能な手段を含む、ユーザ端末が提供される。

【0010】

本発明の第4の態様によれば、処理装置によって読み取り可能なプログラム記憶装置が提供され、前記プログラム記憶装置は、本発明の第1の態様の方法のステップを実行するために前記処理装置により実行可能な命令のプログラムを実施する。

【0011】

50

本発明がより完全に理解されるように、その実施形態を、一例としてのみ、図面を参照して以下に説明する。

【発明を実施するための最良の形態】

【0012】

図1は、本発明の実施形態を組み込んだアーキテクチャを示す。以下で説明するプロセスが実装されてもよいシステム100が示されており、特に、本発明の実施形態のデータベース管理システムによって開始されるデータフローおよびデータ処理ステップが示されている。システム100は、ユーザが対話を行うために使用する、グラフィックユーザインタフェース（GUI）であるユーザインタフェース10を含む。通常は、ユーザが、スプレッドシートレイヤ16にビュー12を入力する。スプレッドシートレイヤ16は、データの
10
スプレッドシート表現を生成するためのレイヤである。スプレッドシートレイヤ16は、スプレッドシートの形式またはそれに類似した形式でデータを表現するためのソフトウェアを含む。スプレッドシートロジックは、ディスプレイ上の行および列内にデータを表示するプログラムである。これは、例えば、経理や簿記などのためのさまざまな用途を有する。一般に、このプログラムは、セルの行および列内の数値および文字列データを操作することが可能である。スプレッドシートプログラムのその他の一般的な機能には、セル内の値が、他のセルを含むことが可能な式から計算されることが可能であるということや、値が、その値が依存する値が変化すると自動的に再計算されてもよいということや、異なるセルが、異なる書式で表示されてもよいということなどが、これらに限定されずに
20
含まれる。代表的なスプレッドシートレイヤとしては、マイクロソフト（Microsoft）社のエクセル（Excel）、ロータス1-2-3（Lotus 1-2-3）などがある。本発明は、使用される特定のスプレッドシートプログラムに関して限定されない。

【0013】

ビューは、データに関する要求を表す。例えば、ビューは、データベースレイヤ28からスプレッドシートレイヤへのデータの配送を発生させてもよく、あるいは、スプレッドシート内のデータがデータベースレイヤ28に保持されるようにしてもよい。ビューは、ユーザにより提供されてもよく、または、システムにより生成されてもよい。ビュー12は、特定のデータが表現または表示されるようにするための要求を含んでいてもよい。ビューを提供した後、ユーザは、対応する問合せの結果を表現するアプリケーションサーバ16からの、データ14の配送という形式で出力を受け取る。スプレッドシートレイヤ16は、前記表現（特に、例えばスプレッドシートなどの、所望の形式でのデータの提供）を実行する、スプレッドシートロジックを実行する。システム100は、さらに、スプレッドシートレイヤ16とデータベースレイヤ28とをインタフェースするための、インタフェースレイヤ22も含む。データベースレイヤ28は、例えばビジネスアプリケーションデータベースなどの、データベース29を含む。データベースは、一般に、それぞれ複数のテーブル内の複数のエンティティタイプの複数のエンティティに対応して、データが記憶され、複数のエンティティが、所定の構造内の複数の関係によってリンクされた、リレーショナルデータベースである。インタフェースレイヤ22は、（それぞれのテーブル内の）異なるエンティティタイプに属するデータを記憶し、そして、それらのタイプのエンティティ間の関係を意味ネットとして記憶するための、意味ネットサーバ23を含む。
30
データベースコンテンツ間の関係の意味モデルを提供し、それと同時に、さまざまなリポジトリからのコンテンツを統合するアクセスレイヤとして役立つ、意味ネット（オントロジ）インフラストラクチャを、データベースコンテンツの上に提供することにより、表現プログラムへの汎用インタフェースが実現される。前記関係は、例えば、基礎をなすデータベース内の外部キー参照により記憶されてもよい。一実施形態では、規定されたビューに従ってデータを収集するために、データベースに対して問合せが実行され、それらのデータが、スプレッドシート内で、特定の構造内に記入される。代替の実施形態では、記入されたスプレッドシート内のデータが読み出されて、そのコンテンツがデータベース内に書き込まれる。データベース29は、構造30およびコンテンツ31を有する。示されて
40
50

いる例では、1つのデータベース29を含む1つのデータベースレイヤのみが示されている。インタフェースレイヤは、少なくとも1つのデータベースとインタフェースするように構成されるため、本発明はこの点で限定されない。例えば、複数のデータベースが、意味ネットサーバ23によってインタフェースされてもよい。データは、矢印18、20により示されるように、スプレッドシートレイヤ16とインタフェースレイヤ22との間で交換されてもよい。同様に、データは、矢印26、27により示されるように、インタフェースレイヤ22とデータベースレイヤ28との間で交換されてもよい。

【0014】

本発明の一実施形態では、データベースの構造30が、意味ネットサーバ23内の意味ネットの中に記憶されたオントロジ内の、項および関係タイプにマッピングされる。好ましくは、データベースコンテンツ31が、項の副項にマッピングされる。さらに、システム上のユーザにより定義されるビュー（当技術分野では問合せとも呼ばれる）は、オントロジの、項および関係タイプ内の意味ビューとして、動的に定義される。好ましい実施形態では、データベース内で持続されているデータベースコンテンツが、最初に、オントロジ（意味ネット構造）内で複製されてもよい。定義された意味ビューが、次に、意味ネットサーバ23内に記憶された意味ネット構造からデータを収集するために使用される。前記複製、およびそれと同時に実行される、データベースコンテンツのマッピングは、一度だけ実行されてもよく、あるいは間欠的に実行されてもよい。例えば、夜間のバッチジョブとして定期的に実行されてもよい。これにより、複製されるデータは最新に維持される。

【0015】

代替の実施形態では、エンティティのための意味ネット構造そのものが必要とされないようにするために、意味ネット構造は、エンティティタイプについてのみ維持され、そして、前記ビューをテクニカル（SQL）ビューに変換し、要求された構造内にその結果を置くために使用できるマッピング情報と関連付けられる。

【0016】

この実施形態では、テクニカルビューは、必要とされるデータのみを、データベースから収集する。収集されたデータは、次に、スプレッドシートに記入するために、意味ネットサーバ23内に記憶された意味ネットの項および関係にマッピングされる。意味問合せは、例えば、スプレッドシート内のタイトル行を記入することにより、スプレッドシートレイヤ16内で定義されてもよく、前記スプレッドシートは、各セル内に、それぞれのタイトル行セルの下のセルに書き込む項目または前記セルから読み出す項目の、関係名（データベース内の1つまたは複数の外部キー参照に対応）および／または意味項（1つまたは複数のデータベースエンティティタイプに対応）を含む。タイトル行の例としては、
「顧客／購入済み、以下の製品を／在住、以下の市に」、
「顧客／購入済み／在住」
「顧客／製品／市」などがある。

【0017】

さらなる実施形態では、意味ネットインフラストラクチャを提供することにより、スプレッドシートプログラム17への汎用インタフェースが、以下の方法で実行されてもよい。スプレッドシートアプリケーション（プログラム）17へのユーザインタフェース10内で、ユーザは、タイトル行を入力し、ここで、各セルは、それぞれの列内に示される要素のエンティティタイプを含む。第1のセルは、デフォルトでは「マスタ」エンティティであり、これは、各行内で、後続セル、従属セルにより含まれている属性または関係が、第1のセル内で見いだされるオブジェクトを参照するということを意味している。マスタセルに続くセル内では、セルのコンテンツと、対応するマスタセル内のエンティティとの間の関係のタイプが、その関係タイプがあいまいでなければ、エンティティタイプの前に指定される。

【0018】

例えば、一実施形態では、タイトル行は次のようになっていてもよい。

【0019】

10

20

30

40

50

【表 1】

タスク	参照、以下の関数を	処理される、 以下の人により	属性を有する、 以下のステータス
-----	-----------	-------------------	---------------------

【0020】

データベースからスプレッドシートアプリケーションへのデータのエクスポートのためには、タイトル行が、意味ネットモデルを含むビジネスアプリケーションに転送される。これは、例えば、ビジネスアプリケーションデータベース29をクリップボードにコピーすることにより実行される。次に、マスタタイプのエンティティの組が選択され、エクスポートが起動される。上の例では、マスタタイプは「タスク」である。タイトル行が解析され、次に、第1列に、選択されたエンティティが記入される。その他の列には、それぞれの属性または関係が記入される。

10

【0021】

関係タイプ「参照」「処理される」「属性を有する」は、マスタエンティティタイプとそれぞれのセルのエンティティタイプとの間に1種類の関係のみがある場合、または属性名があいまいでない場合は、省略されてもよい。

【0022】

ビジネスアプリケーションデータベース29内への、スプレッドシート内の変更のインポートのためには、タイトル行を含むスプレッドシート全体がビジネスアプリケーションに転送される。再び、クリップボードが、これを行うための便利な方法を代表する。タイトル行を解析した後は、スプレッドシートの構造がアプリケーションに知られているため、コンテンツが、最初に、意味ネット表現と整合させられることが可能であり、実装によっては、2番目のステップとして、またはただちに、それぞれのリポジトリに変更が書き込まれてもよい。

20

【0023】

従属するセル内のエンティティの属性または関係を含む、より複雑なビュー（やはり、当技術分野では問合せと呼ばれる）を処理するために、従属タイトルセル内に、この特定の従属セルのコンテンツが、後続するセルのためのマスタと見なされるべきであることを示すマークが含まれていてもよい。元のマスタが、後続するセルのコンテンツのために再び有効と見なされるべきであることを示すために、さらなるマークが定義されてもよい。

30

【0024】

次に例を示す。

【0025】

【表 2】

タスク	関数	(人	部門)	ステータス
-----	----	----	-----	-------

40

【0026】

この例では、「人」の前の括弧は、後続するセル（この例では「部門」）が、タスクではなく、行内の人を参照するということを示す。「部門」の後の閉じ括弧は、元のマスタセル（この例では「タスク」）が、後続するセルのために再び有効であることを示す。さらなる実施形態では、これらの構造はネストされていてもよい。

【0027】

本発明の一実施形態では、データを記憶するためのデータベースを含むデータベースレイヤ28と、ビューに従ってデータベースレイヤ28からのデータをスプレッドシート内

50

で表現するためのスプレッドシートレイヤ16とを備えたデータベース管理システムが提供され、ここで、ビューは、データベースレイヤからスプレッドシートレイヤへのデータの配送、およびスプレッドシートレイヤからデータベースレイヤへのデータの保持のうちの、少なくとも1つを発生させ、そして、前記システムはさらに、データベースレイヤ28とスプレッドシートレイヤ16とをインタフェースするように構成されたインタフェースレイヤ22を備え、ここで、データベースレイヤ28は、それぞれ複数のテーブル内の複数のエンティティタイプの複数のエンティティに対応するデータを、データベースに記憶し、ここで、複数のエンティティは、所定の構造内の複数の関係によりリンクされており、ここで、インタフェースレイヤ22は、データベースレイヤ28内のデータの構造を意味ネットに変換するための意味ネットサーバ23を備え、ビューは、データベースレイヤ28とスプレッドシートレイヤ16との間の相互運用性を容易にするために、意味ネットに関して定義される。これにより、平均的なユーザは、データベースの内部構造の知識なしに、そして、プログラミング技術なしに、最小の努力で、ビジネスアプリケーションデータベースのコンテンツについての任意のビューを表すスプレッドシートを得ることができる。

10

20

30

40

50

【0028】

本発明は、1つまたは複数のスプレッドシートレイヤに適用可能である。システム100に含まれるスプレッドシートレイヤまたはアプリケーションによっては、システムは複数のGUIおよび関連するスプレッドシートレイヤを含んでいてもよいことが理解される。図1には、1つのGUIおよびスプレッドシートレイヤ16のみが示されているが、その他のGUIおよびスプレッドシートレイヤもシステムに含まれていてもよい。インタフェースは、複数のスプレッドシートレイヤと複数のデータベースレイヤとをインタフェースすることが可能である。好ましい実施形態では、インタフェース22は、ビューにตอบสนองして、データベース29内のデータ31がスプレッドシート内で表現されるのを容易にするように適合される。これにより、データ31は、データベース29からスプレッドシート表現17にエクスポートされることが可能となる。さらに、インタフェースレイヤは、データベース29内に記憶される表現でデータが表現されるように構成されていてもよい。これにより、データは、スプレッドシートからデータベース29にエクスポートされることが可能となる。さらに、入力により、スプレッドシート内のデータに変更が発生した場合、データの任意の変更は、データベース29内に記憶されてもよい。これにより、ユーザがデータベースプログラミング言語の知識を有する必要なしに、スプレッドシート内で行われた変更がデータベース29にエクスポートされることが可能となる。特に、意味ネットサーバ23は、データベースレイヤ28内のデータ31の間の関係の意味ネットを記憶する。

【0029】

さらなる実施形態では、ビューは、データベース29内のデータがデータを収集するように定義され、収集されたデータは、所定の構造で、スプレッドシート内で表現される。特に、意味ネットサーバ23は、データベース29の構造から、意味ネット内の複数の項および関係タイプへの、マッピングを記憶する。さらに、意味ネットサーバ23は、データベース29のコンテンツを表すデータから、複数の項にそれぞれ対応する複数の副項への、マッピングを記憶する。また、ビューは、意味ネットの複数の項および関係タイプに関して定義されてもよい。さらなる実施形態では、意味ネットサーバ23は、意味ネット内で、データベース29のデータコンテンツを複製するように構成されている。好ましくは、インタフェースレイヤ22は、ビューを解釈し、ビューに従って、意味ネットからの特定のデータを識別するように構成されている。

【0030】

代替の実施形態では、インタフェースレイヤ22は、ビューを、データベースサーバ28互換ビューに変換するように構成され、ここで、データベースサーバ互換ビューは、データベース29からのデータを収集するために使用される。さらに、意味ネットサーバ23は、収集されたデータを、意味ネットの項および関係にマッピングするように構成され

、ここで、マッピングされたデータはスプレッドシート内に含まれる。

【0031】

上述のように、前記表現は、少なくともタイトル行を含むスプレッドシートであってもよく、ここで、ビューは、タイトル行にビューを入力することにより定義され、ここで、タイトル行は、それぞれ特定の関係に関連する複数のセルのうちの少なくとも1つを含み、ここで、ビューの関係は、1つまたは複数の外部キー参照に対応し、ビューの意味項は、1つまたは複数のデータベースエンティティタイプに対応している。

【0032】

さらなる実施形態では、ビューは、データを収集するために、データベースに対する問合せを実行するために使用され、それらのデータは、所定の構造に従って、スプレッドシートレイヤ17により生成されたスプレッドシート内にインポートされる。これにより、データベース29の構造と前記表現とは関連していない場合でも、データベースの構造の知識をユーザが必要とすることなしに、それらの間で容易にデータが転送されることが可能となる。さらに、スプレッドシートレイヤにより生成された表現で表されたデータが、インタフェースレイヤ22を介して、データベース29にインポートされる。

【0033】

図2は、データベースを管理する方法を組み込んだフロー図を示し、前記方法は、データベースレイヤ28内のデータベース29内にデータを記憶するステップと、ビューに従ってデータベースレイヤからのデータをスプレッドシート内で表現するステップとを含み、ここで、ビューは、データベースレイヤからスプレッドシートレイヤへのデータの配送、およびスプレッドシートレイヤからデータベースレイヤへのデータの保持のうちの、少なくとも1つを発生させ、前記方法はさらに、データベースレイヤ28とスプレッドシートレイヤ16とをインタフェースするステップを含み、ここで、データベースレイヤ28は、それぞれ複数のテーブル内の複数のエンティティタイプの複数のエンティティに対応するデータを、データベースに記憶し、ここで、複数のエンティティは、所定の構造内の複数の関係によりリンクされており、ここで、前記インタフェースするステップは、データベースレイヤ28内のデータの構造を意味ネットに変換するステップを含み、前記表現するステップは、データベースレイヤ28とスプレッドシートレイヤ16との間の相互運用性を容易にするために、意味ネットに関して定義されたビューを入力するステップを含む。

【0034】

図からわかるように、示されているプロセス200は、所定の構造内の複数の関係によりリンクされた複数のエンティティに対応するデータを、データベース内に記憶するステップ（ステップ40）と、データベース内のデータの構造の意味ネットを、意味ネットサーバ内に記憶するステップ（ステップ42）と、ビューを受け取るようにスプレッドシートレイヤを構成するステップ（ステップ44）と、ビューに回答してスプレッドシートとしてのデータの表現を生成するステップ（ステップ46）とを含み、ここで、ビューは、データベースとスプレッドシートレイヤとの間の相互運用性を容易にするために、意味ネットに関して定義される。

【0035】

前記方法は、さらに、ビューに回答して、データベース29内のデータ31がスプレッドシート内で表されるように、インタフェースレイヤ22を構成するステップも含んでもよい。代替の実施形態は、スプレッドシート内で表現されたデータが、インタフェースレイヤ22を介してデータベース29内に記憶されるように、インタフェースレイヤ22を構成するステップを含む。さらに、入力により前記表現内のデータを変更させた場合は、データの任意の変更が、インタフェースレイヤ22を介してデータベース29内に記憶されてもよい。さらなる実施形態では、データベースレイヤ28内のデータ31間の関係の意味ネットを記憶するように、意味ネットサーバ23を構成するステップが含まれる

。さらなる実施形態は、データがスプレッドシートレイヤに配送され、所定の構造でスプレッドシート内で表現されるように、インタフェースレイヤ22を介して、ビューに従って、データベースレイヤ28内のデータに対する問合せを実行するステップを含む。データベースの構造から、意味ネット内の複数の項および関係タイプへのマッピングを記憶するように、意味ネットサーバを構成する、さらなるステップが実行されてもよい。

【0036】

さらに、データベースのコンテンツを表すデータから、複数の項にそれぞれ対応する複数の副項へのマッピングを記憶するように、意味ネットサーバを構成するステップが実行されてもよい。

【0037】

前記方法は、また、意味ネットの複数の項および関係タイプによってビューを定義するステップも含んでいてもよい。前記方法は、さらに、データベースレイヤ28のデータコンテンツ31を意味ネット内で複製するように、意味ネットサーバを構成するステップも含んでいてもよい。ビューを解釈し、ビューに従って意味ネットからの特定のデータを識別するように、インタフェースレイヤ22を構成する、追加のステップも実行されてもよい。

【0038】

ビューをデータベースサーバ互換ビューに変換するようにインタフェースを適合させる、さらなるステップも実行されてもよく（ここで、データベースサーバ互換ビューは、データベースからのデータを収集する）、同様に、収集されたデータを意味ネットの項および関係にマッピングするように、意味ネットサーバ23を構成する、さらなるステップも実行されてもよい（ここで、マッピングされたデータはスプレッドシート内に含まれる）。

【0039】

具体的な実施形態では、少なくともタイトル行を含む、スプレッドシートを提供するステップが実行される。さらに、タイトル行内にビューを入力することによりビューを定義するステップが実行され、ここで、タイトル行は、それぞれ特定の関係に関連する複数のセルのうちの少なくとも1つを含み、ここで、関係は、1つまたは複数の外部キー参照に対応し、意味項は、1つまたは複数のデータベースエンティティタイプに対応している。

【0040】

さらなる実施形態は、データを収集するために、データベースに対して、ビューに従った問合せを実行するさらなるステップを含み、ここで、収集されたデータは、前記タイトル行により決定された構造に従って、スプレッドシートレイヤにより生成された表現内にインポートされる。インタフェースを介して生成された表現で表されたデータを、データベースにインポートするステップも実行されてもよい。

【0041】

本発明は、上述の任意の方法を実行するために動作可能な手段を含む、ユーザ端末への適用例を有する。本発明は、処理装置によって読み取り可能なプログラム記憶装置への適用例を有し、前記プログラム記憶装置は、上述の任意の方法のステップを実行するために前記処理装置により実行可能な命令のプログラムを実施する。本発明は、1つまたは複数の適用例内で実施されてもよい。例えば、本発明は、データベースレイヤ、インタフェースレイヤ、およびスプレッドシートレイヤを組み込んだ1つの適用例内で実施されてもよい。あるいは、各レイヤが、別個の適用例により実施されてもよい。あるいは、1つの適用例が、いくつかのレイヤ（ただし、すべてのレイヤではない）を実施してもよい。本発明は、この点に限定されない。

【0042】

説明した本発明は、ディジタル電子回路内に実装することも、コンピュータハードウェア、ファームウェア、ソフトウェア内に実装することも、それらの組み合わせにより実装することも可能である。本発明による装置は、プログラマブルプロセッサによる実行のために機械読み取り可能な記憶装置内に有形に実施されたコンピュータプログラム製品内に

10

20

30

40

50

実装することが可能であり、本発明の方法のステップは、入力データへの操作を行って出力を生成することにより本発明の機能を実行するための、命令のプログラムを実行するプログラマブルプロセッサにより実行されることが可能である。本発明は、データ記憶装置、少なくとも1つの入力装置、および少なくとも1つの出力装置に、それらとの間でデータと命令を受け渡しするために接続された、少なくとも1つのプログラマブルプロセッサを含む、プログラム可能なシステム上で実行可能な、1つまたは複数のコンピュータプログラム内に実装することができる。各コンピュータプログラムは、高レベルの手続き型言語またはオブジェクト指向プログラミング言語内に、あるいは所望される場合は、アセンブリ言語または機械語内に実装されることが可能であり、いずれの場合も、言語はコンパイルされた、または解釈実行された言語であってもよい。好適なプロセッサには、一例として、汎用マイクロプロセッサおよび専用マイクロプロセッサの両方が含まれる。一般に、プロセッサは、命令とデータを、読み取り専用メモリおよび／またはランダムアクセスメモリから受け取る。通常、コンピュータは、データファイルを記憶するための1つまたは複数の大容量記憶装置を含み、そのような装置としては、内蔵ハードディスクやリムーバブルディスクなどの磁気ディスク、光磁気ディスク、および光ディスクなどがある。コンピュータプログラム命令およびデータを有形に実施するための好適な記憶装置としては、EEPROMやフラッシュメモリ装置などの半導体メモリ装置、内蔵ハードディスクやリムーバブルディスクなどの磁気ディスク、光磁気ディスク、およびCD-ROMディスクを一例として含む、あらゆる形態の不揮発性メモリがある。前述のいずれも、ASIC（特定用途向け集積回路）によって補足されること、またはASIC内に組み込まれることが可能である。

【0043】

ユーザとの対話を提供するために、本発明は、ユーザに情報を表示するためのモニタまたはLCD画面などの表示装置と、ユーザがコンピュータシステムに入力を提供することを可能にする、キーボード、およびマウスまたはトラックボールなどのポインティングデバイスとを有するコンピュータシステム上に実装できる。コンピュータシステムは、コンピュータプログラムがユーザと対話するための、グラフィカルユーザインタフェースを提供するようにプログラムされることが可能である。

【0044】

上記では、本発明の特定の実施形態について説明したが、本発明は、説明した以外の形態で実施されてもよいことがわかるであろう。上記の説明は、本発明を限定することを意図したものではない。例えば、本発明の手順が実行される順序は、請求項内で規定された順序に限定されない。

【図面の簡単な説明】

【0045】

【図1】本発明の一実施形態を組み込んだアーキテクチャを示す。

【図2】本発明の一実施形態に従ってデータを管理する方法を組み込んだフロー図を示す。

【符号の説明】

【0046】

- 10 ユーザインタフェース
- 12 ビュー
- 14 スプレッドシート、データ
- 16 スプレッドシートレイヤ、アプリケーションサーバ
- 17 スプレッドシートプログラム、スプレッドシートアプリケーション、スプレッドシート表現
- 18 ビュー
- 20 データ
- 22 インタフェースレイヤ
- 23 意味ネットサーバ

10

20

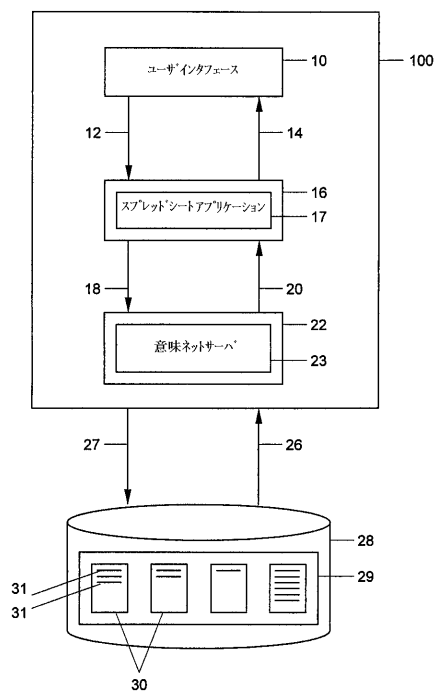
30

40

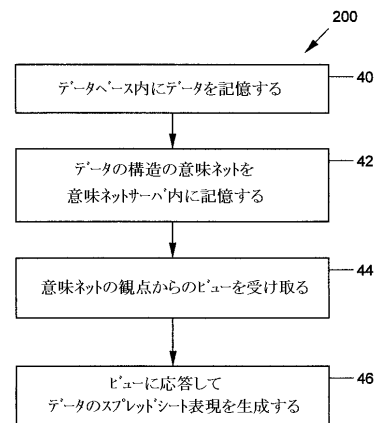
50

- 26 データ
- 27 データベースサーバ互換ビュー
- 28 データベースレイヤ
- 29 データベース、ビジネスアプリケーションデータベース
- 30 構造
- 31 データ、コンテンツ
- 40、42、44、46 ステップ
- 100 システム
- 200 プロセス

【図 1】



【図 2】



フロントページの続き

Fターム(参考) 5B075 NR03
5B082 GA08

【外国語明細書】

Title: A database management system and a method of managing a database

Technical Field

The invention relates to a database management system and a method of managing a database.

Background to the Invention and Prior Art

Databases and other repositories for data are well known for storing data. Further, applications for displaying data in a spreadsheet, or similar, representation are known. Such applications are referred hereinafter as representation applications.

One problem with conventional representation applications is that it is not possible for a user to transfer the contents of a database to a spreadsheet and vice versa without in depth knowledge the relevant database query language, for example, structured query language (SQL) and the internal data structure of the database. Conventionally, a structured query language (SQL) query for a desired view on the database contents is defined. Further a hard-coded program creating a spreadsheet from the view contents is defined. Such a view cannot be defined by a typical user since it requires knowledge of SQL and the internal data structure of the database. Furthermore, it is very complex to create a view that integrates data from different databases or other repositories.

It is an object of the present invention to address those problems encountered with conventional databases and representation applications or spreadsheet layers which are part of a database application. For the sake of simplicity, both separate spreadsheet applications and spreadsheet layers integrated into other applications are referred hereinafter as spreadsheet layers.

In particular, it is an object of the present invention to transfer database contents to a spreadsheet representation and, alternatively, to incorporate changes which are made in the spreadsheet in the original database.

Summary of the Invention

According to a first aspect of the invention, there is provided a database management system comprising a database layer including a database for storing data, a spreadsheet layer for representing data from the database layer in a spreadsheet in accordance with a view, wherein the view causes at least one of a delivery of data to the spreadsheet layer from the database layer and a maintenance of data from the spreadsheet layer to the database layer, the system further comprising an interface layer arranged to interface the database layer with the spreadsheet layer, wherein the database layer stores in the database data corresponding to a plurality of entities of a plurality of entity types in a plurality of tables, respectively, wherein the plurality of entities are linked by a plurality of relations in a predetermined structure, wherein the interface layer comprises a semantic net server for transforming the structure of the data in the database layer into a semantic net and the view is defined in terms of the semantic net to facilitate an interoperability between the database layer and the spreadsheet layer.

By providing a semantic net infrastructure on top of the database contents, the present invention allows database contents to be transferred to a representation, for example, a spreadsheet without the user having to have any knowledge of the database structure or query language. A further advantage of the present invention is the spreadsheet layer does not need to be modified since the exchange of view and result may be done completely via a clipboard.

According to a second aspect of the present invention, there is provided a method of managing a database comprising the steps of:
storing data in a database in a database layer,
representing data from the database layer in a spreadsheet in accordance with a view, wherein the view causes at least one of a delivery of data to the

spreadsheet layer from the database layer and a maintenance of data from the spreadsheet layer to the database layer,
the method including the further steps of:
interfacing the database layer with the spreadsheet layer, wherein the database layer stores in the database data corresponding to a plurality of entities of a plurality of entity types in a plurality of tables, respectively, wherein the plurality of entities are linked by a plurality of relations in a predetermined structure, wherein the interface step includes the step of transforming the structure of the data in the database layer into a semantic net and the representing step includes the step of inputting the view defined in terms of the semantic net, to facilitate an interoperability between the database layer and the spreadsheet layer.

According to a third aspect of the present invention, there is provided a user terminal comprising means operable to perform any of the methods of the first aspect of the present invention.

According to a fourth aspect of the present invention, there is provided a program storage device readable by a processing apparatus, said device embodying a program of instructions executable by the processor to perform the steps of the methods of the first aspect of the present invention.

Brief description of the drawings

In order that the invention may be more fully understood embodiments thereof will now be described by way of example only, with reference to the figures in which:

Figure 1 shows an architecture incorporating an embodiment of the present invention; and

Figure 2 shows a flow diagram incorporating a method of managing data in accordance with an embodiment of the present invention.

Description of the Preferred embodiments

Figure 1 shows an architecture incorporating an embodiment of the present invention. A system 100 is shown on which the process described hereinafter may be implemented, in particular the data flow and data processing steps initiated by the database management system of an embodiment of the present invention are shown. The system 100 includes a user interface 10 which is a graphic user interface (GUI), with which a user interacts. Typically, the user inputs a view 12 to a spreadsheet layer 16. The spreadsheet layer 16 is a layer for generating a spreadsheet representation of data. The spreadsheet layer 16 includes software for representing data in the form of a spreadsheet or the like. Spreadsheet logic is a program that displays data in rows and columns on a display. It has a variety of uses, for example, for accounting or bookkeeping. Typically, the program is able to manipulate numerical and string data in rows and columns of cells. Other typical features of the spreadsheet program include, but are not limited to: the value in a cell can be calculated from a formula which can involve other cells, a value may be recalculated automatically whenever a value on which it depends changes, different cells may be displayed with different formats, etc. Typical spreadsheet layers include Microsoft's Excel, Lotus 1-2-3, etc. The invention is not limited with respect to the particular spreadsheet program used.

A view represents a request with respect to data. For example, it may cause delivery of data from a database layer 28 to the spreadsheet layer, alternatively, it may cause data in the spreadsheet to be maintained to the database layer 28. A view may be provided by a user or generated by the system. The view 12 may include a request to have certain data represented or displayed. Having provided a view, the user receives an output in the form of a delivery of data 14 from the application server 16 representing the results of the corresponding query. The spreadsheet layer 16 runs the spreadsheet logic which carries out the representation, in particular, the provision of the data in

the desired format, for example, a spreadsheet. The system 100 further includes an interface layer 22 for interfacing the spreadsheet layer 16 and a database layer 28. The database layer 28 includes a database 29, for example, a business application database. The database is typically a relational database wherein data is stored corresponding to a plurality of entities of a plurality of entity types in a plurality of tables, respectively, wherein the plurality of entities are linked by a plurality of relations in a predetermined structure. The interface layer 22 includes a semantic net server 23 for storing the data belonging to different entity types (in respective tables) and relations between entities of these types as a semantic net. By providing a semantic net (ontology) infrastructure on top of the database contents, which provides a semantic model of the relations between database contents and at the same time serves as an access layer that integrates contents from different repositories, a generic interface to a representation program is realised. The relations may, for example, be stored by foreign key references in the underlying databases. In one embodiment a query is performed on the database to collect data according to the defined view and these data are filled into a spreadsheet in a certain structure. In an alternative embodiment, the data in a filled-in spreadsheet is read and its contents written into the database. The database 29 has a structure 30 and contents 31. In the example shown only one database layer including one database 29 is shown. The invention is not limited in this respect since the interface layer is arranged to interface at least one database. For example, a plurality of databases may be interfaced by the semantic net server 23. Data may be exchanged between the spreadsheet layer 16 and the interface layer 22 as shown by arrows 18, 20. Similarly, data may be exchanged between the interface layer 22 and the database layer 28, as shown by arrows 24, 26.

In an embodiment of the present invention, the structure 30 of the database is mapped onto terms and relation types in an ontology stored in a semantic net within the semantic net server 23. Preferably, the database contents 31 are

mapped onto subterms of the terms. Further, the view (also referred to in the art as a query) defined by a user on the system is dynamically defined as a semantic view in terms and relation types of the ontology. In a preferred embodiment, the database contents may first be replicated in an ontology (semantic net structure) which is persisted in a database. The semantic view, that is as defined, is then used to collect the data from the semantic net structure as stored in the semantic net server 23. The replication and at the same time mapping of the database contents may be done only once, or alternatively intermittently. For example, on a regular basis as an overnight batch job. In this way, the replicated data is kept up to date.

In an alternative embodiment, the semantic net structure is persisted only for the entity types and associated with mapping information that can be used to translate the view into a technical (SQL) view and put the results into the required structure so that a semantic net structure for the entities themselves is not needed.

In this embodiment, the technical view collects only the required data from the database or databases. The collected data is then mapped onto the semantic net terms and relations as stored in the semantic net server 23 to fill in the spreadsheet. The semantic query may be defined in the spreadsheet layer 16, for example, by filling in a title row in the spreadsheet, which contains in each cell a relation name corresponding to one or more foreign key references in the database(s) and/or the semantic term corresponding to one or more database entity types, of the item to write in or be read from the cell under the respective title row cell. Examples for the title row include:

"customer / has purchased product / lives in city",

"customer / has purchased / lives in",

"customer / product / city".

In a further embodiment, by providing a semantic net infrastructure, a generic interface to a spread sheet program 17 may also be carried out in the following

way. In the user interface 10 to the spreadsheet application 17, the user enters a title line where each cell contains the entity type of the elements to be shown in the respective column. The first cell is by default the "master" entity, which means that in each line, the attributes or relations contained by the following cells, the dependent cells, refer to the object found in the first cell. In the cells following the master cell, the type of relation between the cell content and the entity in the corresponding master cell is specified before the entity type if that relation type is not unambiguous.

For example, in one embodiment, a title line may read as follows:

Task	refers to function	handled by person	has attribute Status
------	--------------------	----------------------	-------------------------

For an export of data from the database to the spreadsheet application, the title line is transferred to the business application containing the semantic net model. This is carried out, for example, by copying the business application database 29 onto a clipboard. Then a set of entities of the master type is selected and the export is triggered. In the example above, the master type is "Task". The title line is parsed and then the first column is filled with the selected entities. The other columns are filled with the respective attributes or relations.

The relation types: "refers to", "handled by", "has attribute", can be left out if there is only one kind of relation between the master entity type and the entity type of the respective cell or if the attribute name is unambiguous.

For an import of changes in a spreadsheet into the business application database 29, the whole spread sheet including the title line is transferred to the business application. Again, the clipboard represents a convenient way of doing this. After parsing the title line, the structure of the spreadsheet is known to the application and the contents can be first aligned with the semantic net representation and, in a second step or immediately, depending

on the implementation, the changes may be written to the respective repositories.

In order to handle more complex views (also referred to as queries in the art), including attributes or relations of entities in dependent cells, a mark may be included in a dependent title cell, indicating that the content of this particular dependent cell is to be considered as the master for the following cell. A further mark may be defined to indicate that the original master is to be considered valid for the contents of the following cells again.

For example,

Task	function	(person	department)	Status
------	----------	---------	-------------	--------

In this example, the parenthesis before "person" indicates that the following cells, in this example "department", refer to the person in the line, rather than the task. The closing parenthesis after "department" indicates that the original master cell, in the example "task", is valid again for the following cells. In a further embodiment, these structures may be nested.

In one embodiment of the present invention, there is provided a database management system comprising a database layer 28 including a database for storing data, a spreadsheet layer 16 for representing data from the database layer 28 in a spreadsheet in accordance with a view, wherein the view causes at least one of a delivery of data to the spreadsheet layer from the database layer and a maintenance of data from the spreadsheet layer to the database layer, the system further comprising an interface layer 22 arranged to interface the database layer 28 with the spreadsheet layer 16, wherein the database layer 28 stores in the database data corresponding to a plurality of entities of a plurality of entity types in a plurality of tables, respectively, wherein the plurality of entities are linked by a plurality of relations in a predetermined structure, wherein the interface layer 22 comprises a semantic

net server 23 for transforming the structure of the data in the database layer 28 into a semantic net and the view is defined in terms of the semantic net to facilitate an interoperability between the database layer 28 and the spreadsheet layer 16. In this way the average user is able to derive a spreadsheet representing an arbitrary view on contents of a business application database with minimum effort and without knowledge of the internal structure of the databases and without programming skills.

The present invention is applicable to one or more spreadsheet layers. It will be understood that depending on the spreadsheet layer or applications comprised in the system 100, the system may include a plurality of GUIs and associated spreadsheet layers. Although, only one GUI and spreadsheet layer 16 are shown in Figure 1 other GUIs and spreadsheet layers may also be comprised in the system. The interface being able to interface a plurality of spreadsheet layers with a plurality of database layers. In a preferred embodiment, the interface 22 is adapted to facilitate, in response to the view, the data 31 in the database 29 to be represented in the spreadsheet. In this way, data 31 may be exported from the database 29 to the spreadsheet representation 17. Further, the interface layer may be arranged so that the data represented in the representation to be stored in the database 29. In this way, data may be exported from the spreadsheet to the database 29. Further, if an input causes the data in the spreadsheet to be changed, any change in the data may be stored in the database 29. In this way changes made in the spreadsheet may be exported to the database 29 without the user having to have any knowledge of the database programming language. In particular, the semantic net server 23 stores a semantic net of the relations between the data 31 in the database layer 28.

In a further embodiment, the view is defined for the data in the database 29 to collect data, and the collected data are represented in the spreadsheet in a predetermined structure. In particular, the semantic net server 23 stores a

mapping of the structure of the database 29 onto a plurality of terms and relation types in a semantic net. Further, the semantic net server 23 stores a mapping of the data representing contents of the database 29 onto a plurality of subterms corresponding to the plurality of terms, respectively. Also, the view may be defined in terms of the plurality of terms and relation types of the semantic net. In a further embodiment, the semantic net server 23 is arranged to replicate the data contents of the database 29 in the semantic net. Preferably, the interface layer 22 is arranged to interpret the view and to identify particular data from the semantic net according to the view.

In an alternative embodiment, the interface layer 22 is arranged to translate the view into a database server 28 compatible view, wherein the database server compatible view is used to collect the data from the database 29. Further, the semantic net server 23 is arranged to map the collected data onto the terms and relations of the semantic net, wherein the mapped data is included in the spreadsheet.

As mentioned, the representation may be a spreadsheet including at least a title row, wherein the view is defined by entering a view in the title row, wherein the title row includes at least one of a plurality of cells which is associated with a particular relation respectively, wherein the relation corresponds to one or more foreign key references, and a semantic term corresponding to one or more database entity types, of the view.

In a further embodiment, the view is used to perform a query on the database to collect data and these data are imported into a spreadsheet generated by the spreadsheet layer 17 according to a predetermined structure. In this way, although the structure of the database 29 and the representation may be unrelated, data may be transferred between them easily without the user requiring knowledge of the structure of the database. Further, the data represented in a representation generated by the spreadsheet layer is imported via the interface layer 22 to the database 29.

Figure 2 shows a flow diagram incorporating a method of managing a database comprising the steps of:

storing data in a database 29 in a database layer 28,

representing data from the database layer in a spreadsheet in accordance with a view, wherein the view causes at least one of a delivery of data to the spreadsheet layer from the database layer and a maintenance of data from the spreadsheet layer to the database layer,

the method including the further steps of:

interfacing the database layer 28 with the spreadsheet layer 16, wherein the database layer 28 stores in the database data corresponding to a plurality of entities of a plurality of entity types in a plurality of tables, respectively, wherein the plurality of entities are linked by a plurality of relations in a predetermined structure, wherein the interface step includes the step of transforming the structure of the data in the database layer 28 into a semantic net and the representing step includes the step of inputting the view defined in terms of the semantic net, to facilitate an interoperability between the database layer 28 and the spreadsheet layer 16.

As can be seen a process 200 is shown which includes the steps of storing data corresponding to a plurality of entities which are linked by a plurality of relations in a predetermined structure in the database (step 40), storing a semantic net of the structure of the data in the database in the semantic net server (step 42), and

arranging the spreadsheet layer to receive a view (step 44) and generate a representation of data as a spreadsheet in response to the view (step 46), wherein the view is defined in terms of the semantic net to facilitate an interoperability between the database and the spreadsheet layer.

The method may include the further step of arranging the interface layer 22 so that, in response to the view, the data 31 in the database 29 is represented in the spreadsheet. An alternative embodiment includes the step of arranging the interface layer 22 so that the data represented in the spreadsheet is stored via

the interface layer 22 in the database 29. Further, if an input causes the data in the representation to be changed, any change in the data may be stored via the interface 22 in the database 29. In a further embodiment, the step of arranging the semantic net server 23 to store a semantic net of the relations between the data 31 in the database layer 28 is included. A further embodiment includes the step of performing a query on the data in the database layer 28, according to the view, via the interface layer 22 so that data is delivered to the spreadsheet layer and represented in the spreadsheet in a predetermined structure. The further step of arranging the semantic net server to store a mapping of the structure of the database onto a plurality of terms and relation types in a semantic net may be carried out.

Also, the step of arranging the semantic net server to store a mapping of the data representing contents of the database onto a plurality of subterms corresponding to the plurality of terms, respectively may be carried out.

The method may also include the step of defining the view by the plurality of terms and relation types of the semantic net. It may further include the step of arranging the semantic net server to replicate the data contents 31 of the database layer 28 in the semantic net. The additional step of arranging the interface layer 22 to interpret the view and to identify particular data from the semantic net according to the view may also be carried out.

The further step of adapting the interface to translate the view into a database server compatible view, wherein the database server compatible view collects the data from the database may also be carried out, as may the further step of arranging the semantic net server 23 to map the collected data onto the terms and relations of the semantic net, wherein the mapped data is included in the spreadsheet.

In a particular embodiment, the step of providing a spreadsheet including at least a title row is carried out. Further, the step of defining the view by entering a view in the title row, wherein the title row includes at least one of a plurality of cells which is associated with a particular relation respectively,

wherein the relation corresponds to one or more foreign key references, and a semantic term corresponding to one or more database entity types, is carried out.

A further embodiment, includes the further step of performing a query according to the view on the database to collect data, wherein the collected data is imported into a representation generated by the spreadsheet layer according to the structure determined by said title row. The step of importing the data represented in a representation generated via the interface to the database may also be carried out.

The present invention has application to a user terminal comprising means operable to perform any of the methods described. The present invention has application to a program storage device readable by a processing apparatus, said device embodying a program of instructions executable by the processor to perform the steps of any one of the methods described. The present invention may be embodied in one or a plurality of applications. For example, the invention may be embodied in a single application incorporating a database layer, an interface layer and a spreadsheet layer. Alternatively, each layer may be embodied by a separate application or applications. Alternatively, a single application may embody some but not all of the layers. The invention is not limited in this respect.

The invention as described can be implemented in digital electronic circuitry, or in computer hardware, firmware, software, or in combination thereof. An apparatus according to the present invention can be implemented in a computer program product tangibly embodied in a machine readable storage device for execution by a programmable processor; and method steps of the invention can be performed by a programmable processor executing a program of instructions to perform functions of the invention by operating on input data and generating output. The invention can be implemented in one or more computer programs that are executable on a programmable system including

at least one programmable processor coupled to receive data and instructions from, and to transmit data and instructions to, a data storage system, at least one input device, and at least one output device. Each computer program can be implemented in a high-level procedural or object-oriented programming language, or in assembly or machine language if desired; and in any case, the language can be a compiled or interpreted language. Suitable processors include, by way of example, both general and special purpose microprocessors. Typically, a processor will receive instructions and data from a read-only memory and/or a random access memory. Typically, a computer will include one or more mass storage devices for storing data files; such devices include magnetic disks, such as internal hard disks and removable disks; magneto-optical disks; and optical disks. Storage devices suitable for tangibly embodying computer program instructions and data include all forms of non-volatile memory, including by way of example, semiconductor memory devices, such as EEPROM, and flash memory devices; magnetic disks such as internal hard disks and removable disks; magneto-optical disks; and CD-ROM disks. Any of the foregoing can be supplemented by, or incorporated in ASICs (application specific integrated circuits).

To provide for interaction with a user, the invention can be implemented on a computer system having a display device such as a monitor or LCD screen for displaying information to the user and a keyboard and a pointing device such as a mouse or a trackball by which the user can provide input to the computer system. The computer system can be programmed to provide a graphical user interface through which computer programs interact with users.

Whilst specific embodiments of the invention have been described above, it will be appreciated that the invention may be practiced otherwise than as described. The description is not intended to limit the invention. For example, the order steps be performed of the present invention is not limited to that order prescribed in the claims.

Claims

1. A database management system comprising a database layer (28) including a database for storing data, a spreadsheet layer (16) for representing data from the database layer (28) in a spreadsheet in accordance with a view, wherein the view causes at least one of a delivery of data to the spreadsheet layer from the database layer and a maintenance of data from the spreadsheet layer to the database layer, the system further comprising an interface layer (22) arranged to interface the database layer (28) with the spreadsheet layer (16), wherein the database layer (28) stores in the database data corresponding to a plurality of entities of a plurality of entity types in a plurality of tables, respectively, wherein the plurality of entities are linked by a plurality of relations in a predetermined structure, wherein the interface layer (22) comprises a semantic net server (23) for transforming the structure of the data in the database layer (28) into a semantic net and the view is defined in terms of the semantic net to facilitate an interoperability between the database layer (28) and the spreadsheet layer (16).
2. A database management system according to claim 1, wherein if the view causes the data in the spreadsheet to be changed, any change in the data (24) is stored in the database layer (28).
3. A database management system according to any of the preceding claims, wherein the semantic net server (23) stores a semantic net of the relations between the data (31) in the database layer (28).

4. A database management system according to any preceding claim, wherein the relations between the plurality of tables are stored as a plurality of foreign key references.
5. A database management system according to any of the preceding claims, wherein the view (12) is performed on the data (31) in the database layer (28) via the interface layer (22) so that data is delivered to the spreadsheet layer and represented in the spreadsheet (14) in a predetermined structure.
6. A database management system according to any of the preceding claims, wherein the semantic net server (23) stores a mapping of the structure (30) of the database (29) onto a plurality of terms and relation types in a semantic net.
7. A database management system according to claim 6, wherein the semantic net server (23) stores a mapping of the data (31) representing a contents of the database (29) onto a plurality of subterms corresponding to the plurality of terms, respectively.
8. A database management system according to claims 6 or 7, wherein the view (24) is defined in terms of the plurality of terms and relation types of the semantic net.
9. A database management system according to any preceding claim, wherein the semantic net server (23) is arranged to replicate the data contents of the database layer (28) in the semantic net.
10. A database management system according to any preceding claim, wherein the interface layer (22) is arranged to interpret the view

and to identify particular data from the semantic net in response to the view.

11. A database management system according to either of claims 9 or 10, wherein the semantic net server (23) is arranged to replicate the data contents (31) of the database layer (28) intermittently.
12. A database management system according to any of the preceding claims, wherein the interface layer (22) is arranged to translate the view (18) into a database server compatible view (24), wherein the database server compatible view collects data (31) in accordance with the view from the database (29).
13. A database management system according to claim 12, wherein the semantic net server (23) is arranged to map the collected data onto the terms and relations of the semantic net, wherein the mapped data is included in the spreadsheet.
14. A database management system according to any of the preceding claims, wherein the spreadsheet representation includes at least a title row, and wherein the view is defined by entering a view in the title row, wherein the title row includes at least one of a plurality of cells which is associated with a particular relation respectively, wherein the relation corresponds to one or more foreign key references, and a semantic term corresponding to one or more database entity types, of the view.
15. A database management system according to any of the preceding claims, wherein a query according to the view is performed on the database (29) to collect data and these data are imported into a

spreadsheet generated by the spreadsheet layer according to a predetermined structure.

16. A database management system according to any of the preceding claims, wherein the data represented in a spreadsheet generated by the spreadsheet layer is imported via the interface layer (22) to the database (29).
17. A method of managing a database comprising the steps of:
storing data in a database (29) in a database layer (28),
representing data from the database layer in a spreadsheet in accordance with a view, wherein the view causes at least one of a delivery of data to the spreadsheet layer from the database layer and a maintenance of data from the spreadsheet layer to the database layer,
the method including the further steps of:
interfacing the database layer (28) with the spreadsheet layer (16),
wherein the database layer (28) stores in the database data corresponding to a plurality of entities of a plurality of entity types in a plurality of tables, respectively, wherein the plurality of entities are linked by a plurality of relations in a predetermined structure, wherein the interface step includes the step of transforming the structure of the data in the database layer (28) into a semantic net and the representing step includes the step of inputting the view defined in terms of the semantic net, to facilitate an interoperability between the database layer (28) and the spreadsheet layer (16).
18. A method according to claim 17, wherein if an input causes the data in the spreadsheet to be changed, any change in the data in the

spreadsheet is stored via the interface layer (22) in the database layer(29).

19. A method according to any of the preceding claims 17 or 18, including the step of arranging the semantic net server (23) to store a semantic net of the relations between the data (31) in the database layer (28).
20. A method according to any of the preceding claims 17-19, including the step of arranging the database layer (28) to store data belonging to different entities in different respective tables and to store relations between entities by foreign key references.
21. A method according to any of the preceding claims 17-20, including the steps of performing the view on the data in the database layer (28) via the interface layer (22) so that data is delivered to the spreadsheet layer and represented in the spreadsheet in a predetermined structure.
22. A method according to any of the preceding claims 17-21, including the further step of arranging the semantic net server (23) to store a mapping of the structure of the database onto a plurality of terms and relation types in a semantic net.
23. A method according to claim 22, including the further step of arranging the semantic net server (23) to store a mapping of the data representing a contents of the database (29) onto a plurality of subterms corresponding to the plurality of terms, respectively.

24. A method according to either of claim 22 or 23, including the further step of defining the view in terms of the plurality of terms and relation types of the semantic net.
25. A method according to any of the preceding claims 17-24, including the further step of arranging the semantic net server (23) to replicate the data contents of the database layer (28) in the semantic net.
26. A method according to any of the preceding claims 17-25, including the step of arranging the interface layer (22) to interpret the view and to identify particular data from the semantic net according to the view.
27. A method according to either of claims 25 or 26, including the further step of arranging the semantic net server (23) to replicate the contents of the database (29) intermittently.
28. A method according to any of the preceding claims 17-27, including the further step of arranging the interface layer (22) to translate the view into a database server compatible view (24), wherein the database server compatible view is used to collect the data from the database (29).
29. A method according to claim 28, including the further step of arranging the semantic net server (23) to map the collected data onto the terms and relations of the semantic net, wherein the mapped data is included in the representation.

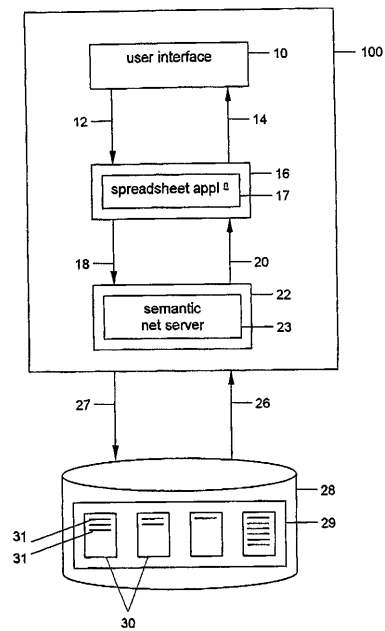
30. A method according to any of the preceding claims 17-29, wherein the spreadsheet representation includes at least a title row, and including the step of defining the view by entering the view in the title row, wherein the title row includes at least one of a plurality of cells which is associated with a particular relation respectively, wherein the relation corresponds to one or more foreign key references, and a semantic term corresponding to one or more database entity types, of the view.
31. A method according to any of the preceding claims 17-30, including the further step of performing a query according to the view on the database (29) to collect data, wherein the collected data is imported into a spreadsheet representation generated by the spreadsheet layer (16) according to a predetermined structure.
32. A method according to any of the preceding claims 17-31, including the step of importing the data represented in a spreadsheet via the interface layer (22) to the database layer (28).
33. A user terminal comprising means operable to perform the method of any of claims 17-32.
34. A program storage device readable by a processing apparatus, said device embodying a program of instructions executable by the processor to perform the steps of any one of claims 17-32.

Abstract

The invention relates to a database management system and a method of managing a database. The database management system comprising a database layer (28) including a database for storing data, a spreadsheet layer (16) for representing data from the database layer (28) in a spreadsheet in accordance with a view, wherein the view causes at least one of a delivery of data to the spreadsheet layer from the database layer and a maintenance of data from the spreadsheet layer to the database layer, the system further comprising an interface layer (22) arranged to interface the database layer (28) with the spreadsheet layer (16), wherein the database layer (28) stores in the database data corresponding to a plurality of entities of a plurality of entity types in a plurality of tables, respectively, wherein the plurality of entities are linked by a plurality of relations in a predetermined structure, wherein the interface layer (22) comprises a semantic net server (23) for transforming the structure of the data in the database layer (28) into a semantic net and the view is defined in terms of the semantic net to facilitate an interoperability between the database layer (28) and the spreadsheet layer (16).

【 図 1 】

Fig.1



【 図 2 】

Fig.2

